Docket No. 42P17831
Express Mail No. EV339914679US

UNITED STATES PATENT APPLICATION

for

AN APPARATUS AND METHOD FOR DISTRIBUTING PRIVATE KEYS TO AN ENTITY WITH MINIMAL SECRET, UNIQUE INFORMATION

Inventor:

Gary L. Graunke

Prepared by:

Blakely Sokoloff Taylor & Zafman, LLP
12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025-1030
(310) 207-3800

AN APPARATUS AND METHOD FOR DISTRIBUTING PRIVATE KEYS TO AN ENTITY WITH MINIMAL SECRET, UNIQUE INFORMATION

FIELD OF THE INVENTION

[0001] One or more embodiments of the invention relate generally to the field of cryptography. More particularly, one or more of the embodiments of the invention relates to a method and apparatus for distributing private keys to an entity with minimal secret, unique information.

BACKGROUND OF THE INVENTION

[0002] The proliferation of the Internet has led to the creation of a new form of commerce, generally referred to as Internet or electronic commerce (E-commerce). E–commerce enables users to sell and purchase items from a worldwide community connected via the Internet. This added simplicity, coupled with the continually reduced costs and increasing processing speed of modern-day computers, has led to the inclusion of a personal computer (PC) in many homes throughout the world. Unfortunately, the proliferation of PCs within the homes throughout the world, as well as the use of such PCs for E-commerce, often results in the storage of sensitive information within a computer.

[0003] As a result, computer users become susceptible to rogue agents, which may desire to gain access to secure information loaded within their personal computer. In order to combat the various rogue agents from gaining access to the secure information, many computer systems employ some form of cryptographs in order to prevent access to sensitive information. As known to those skilled in the art, cryptography provides a technique for keeping information secret, for determining that the information has not been tampered with and for determining who authored pieces of information.

[0004] One form of cryptography involves public/private key systems. Public/private key systems encrypt information prior to transmission using a public key and decrypting received encrypted information using a private key that is only known to the recipient of the encrypted information. However, once the sensitive information arrives at its designated location, the information is often decrypted and stored in a clear format. In other words, the sensitive information is not maintained in a secure format at its destination. As a result, during operation of a PC, a rogue agent could possibly gain access to the PC and gain access to sensitive information.

[0005] Furthermore, the proliferation of E-commerce has led to the availability of media applications, such as motion pictures and music, which may be downloaded to a PC for one-time use or for use for a predetermined period of time. Unfortunately, without some mechanism for protecting the contents of such media applications from access by rogue agents, E-commerce involving media applications may be prohibitive to the media providers. As a result, media or content providers may be reluctant to create high quality media or content providing applications when such content may be susceptible to rogue agents.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] The various embodiments of the present invention are illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which:

[0007] FIG. 1 is a block diagram illustrating a computer system including a chipset having key logic to enable receipt of a private key while storing minimal secret, unique information within the chipset, in accordance with one embodiment.

[0008] FIG. 2 is a block diagram illustrating an overview of distributing private keys to an entity with minimal secret, unique information, in accordance with one embodiment.

[0009] FIG. 3 is a block diagram further illustrating secret key logic of FIG. 2, in accordance with one embodiment.

[00010] FIG. 4 is a block diagram further illustrating key logic of FIG. 1, in accordance with one embodiment.

[00011] FIG. 5 is a block diagram further illustrating key distribution facility of FIG. 2, in accordance with one embodiment.

[00012] FIG. 6 is a flowchart illustrating a method for installing minimal secret, unique information within a manufactured chip to enable distribution of at least one private key to the manufactured chip, in accordance with one embodiment.

[00013] FIG. 7 is a flowchart illustrating a method for installing minimal secret, unique information within a manufactured chip, in accordance with one embodiment.

[00014] FIG. 8 is a flowchart illustrating a method for generating a private key in response to a key update request, in accordance with one embodiment.

[00015] FIG. 9 is a flowchart illustrating a method for authenticating a received key update request, in accordance with one embodiment.

[00016] FIG. 10 is a flowchart illustrating a method for generating a key vector, in accordance with one embodiment.

[00017] FIG. 11 is a flowchart illustrating a method for initializing an integrated chip, including stored secret, unique information, to receive at least one private key, in accordance with one embodiment.

[00018] FIG. 12 is a flowchart illustrating a method for initializing an integrated chip having minimal unique, secret information, in accordance with one embodiment.

[00019] FIG. 13 is a flowchart illustrating a method for initializing an integrated chip having stored minimal secret, unique information to perform authentication using at least one private key assigned to the integrated chip, in accordance with one embodiment.

[00020] FIG. 14 is a flowchart illustrating a method for requesting an integrated chip having stored minimal unique, secret information to perform a key update request, in accordance with one embodiment.

DETAILED DESCRIPTION

[00021] A method and apparatus for distributing private keys to an entity with minimal secret, unique information are described. In one embodiment, the method includes the storage of a chip secret key within a manufactured chip. Once the chip secret key is stored or programmed within the chip, the chip is sent to a system original equipment manufacturer (OEM) in order to integrate the chip within a system or device. Subsequently, a private key is generated for the chip by a key distribution facility (KDF) according to a key request received from the system OEM. In one embodiment, the KDF is the chip manufacturer.

[00022] In the following description, certain terminology is used to describe features of the invention. For example, the term "logic" is representative of hardware and/or software configured to perform one or more functions. For instance, examples of "hardware" include, but are not limited or restricted to, an integrated circuit, a finite state machine or even combinatorial logic. The integrated circuit may take the form of a processor such as a microprocessor, application specific integrated circuit, a digital signal processor, a micro-controller, or the like.

[00023] An example of "software" includes executable code in the form of an application, an applet, a routine or even a series of instructions. The software may be stored in any type of computer or machine readable medium such as a programmable electronic circuit, a semiconductor memory device inclusive of volatile memory (*e.g.*, random access memory, etc.) and/or non-volatile memory (*e.g.*, any type of read-only memory "ROM," flash memory), a floppy diskette, an optical disk (*e.g.*, compact disk or digital video disk "DVD"), a hard drive disk, tape, or the like.

System

[00024] FIG. 1 is a block diagram illustrating computer system 100 including chipset 310 manufactured to include chip secret key (K_c) 250 to enable distribution of a private key to chipset 310 using key logic 320, in accordance with one embodiment. Computer system 100 comprises processor system bus (front side bus (FSB)) 102 for communicating information between processor (CPU) 110 and chipset 310 coupled together via FSB 102. As described herein, the term "chipset" is used to describe, collectively the various devices coupled to CPU 110 to perform desired system functionality.

[00025] Chipset 310 is coupled to main memory 120 and non-volatile (e.g., Flash) memory 150. In one embodiment, main memory 120 is volatile memory including, but not limited to, random access memory (RAM), synchronous RAM (SRAM), double data rate (DDR), synchronous dynamic RAM (SDRAM), rambus dynamic RAM (RDRAM), or the like. In addition, hard disk drive devices (HDD) 130, as well as one or more input/output (I/O) devices 140 (140-1, . . ., 140-N) are also coupled to chipset 310. As illustrated, chipset 310 includes store chip secret key 250 and key logic 320, which are further described with reference to FIG. 2.

[00026] FIG. 2 provides an overview for the installation of chip secret key 250 within chip 222 manufactured by manufacturer 200 and the subsequent generation and assignment of at least one private key 290 to chip 222 once integrated within chipset 310, in accordance with one embodiment. As described herein, chip 222 is may alternatively referred to as manufactured chip 222, and integrated chip 222. In one embodiment, private key 290 is stored within flash memory 150. Representatively, private key 290 enables chip 222 to perform an authentication procedure to establish a secure authenticated channel, in accordance with one embodiment. In one embodiment, a chip secret key 250 enables assignment of at least one public/private key crypto-system key to chip 222.

[00027] In one embodiment, the installation of chip secret key 250 within manufactured chip 222 enables public key cryptography. As described herein, a cryptographic system refers to a system that uses two keys; a public key known to everyone, and a private, or secret, key known only to the recipient of digital content. Accordingly, digital content is initially encrypted by transforming the content into an unreadable format referred to as "cipher text" using a recipient's public key. Subsequently, when the encrypted digital content, or cipher text, is received by the recipient, the received content may be decrypted, or deciphered, using a private key of the recipient to form the digital content in the clear format.

[00028] However, as will be recognized by those skilled in the art, the embodiments described herein are not limited to public key cryptography or asymmetric encryption, which uses a public key and private key pair, but may be used within systems for symmetric encryption, which uses single secret, or private, key. Hence, the techniques described herein can be modified to function within cryptographic system, such as symmetric key systems that use a single key that both the sender and the recipient

have, as well as public key systems that use two public keys; a public key known to everyone and a private key known to only the recipient of encrypted cipher text.

[00029] Referring again to FIG. 2, chip manufacturer 200 initially gathers unique manufacturing information (M_c) for each chip. As illustrated, chip 222 is formed during wafer sort from fabricated wafer 212. Hence, in one embodiment, manufacturing information from each chip may include a wafer serial number from which chip 222 is formed in addition to a coordinate X,Y location of chip 222 within wafer 212. Once this information is formed, the manufacturing information M_c is provided to secret key logic 230, as further illustrated with reference to FIG. 3.

[00030] As illustrated with reference to FIG. 3, manufacturing information M_c 232 is initially provided to first block cipher 236. First block cipher 236 also receives first key (K_1) 234. As illustrated, block cipher 236 encrypts M_c 232 to form a unique chip ID (ID_c) 240. As further illustrated, chip ID 240 is provided to second block cipher 244, which encrypts ID_c 240 to form chip secret key (K_c) 250. Once chip secret key 250 is formed, chip secret key 250 is provided to K_c program logic 252. In one embodiment, program logic 252 installs chip secret key 250 within manufactured chip 222.

[00031] In one embodiment, block cipher 236 and block cipher 244 may be implemented using the advanced encryption standard (AES), the triple data encryption standard (3DES), the data encryption standard (DES) or other like encryption/decryption standard. Accordingly, as described herein, the term cryptographic block refers to logic designed to encrypt content or decrypt cipher text according to AES, DES, 3DES or other like encryption/decryption standard.

[00032] In one embodiment, chip secret key 250 is installed and programmed into manufactured chip 222 by blowing fuses or equivalent mechanism to store chip set key 250 within manufactured chip 222. Once installed, chip 222 is sent to system OEM 300 for integration. For example, referring again to FIG. 3, secret key logic 230 transmits manufactured chip 222, including chip secret key 250 to OEM 300 for integration within chipset 310. Once installed or integrated within chipset 310, OEM 300 initializes chipset 310 to generate key request 322 using key logic 320 (FIG. 1), as further illustrated with reference to FIG. 4.

[00033] As illustrated with reference to FIG. 4, key logic 320 includes a first block cipher 322, which receives cipher text (G) 302 from OEM 300 during initialization of chipset 310. As illustrated, key logic 320 decrypts cipher text G 302 using chip secret

key 250 in order to form chip ID 240. However, as part of the initialization process, OEM 300 initially generates random cipher text G 302, which is to include chip ID 240, at least one private key assigned to chipset 310 and a private key digital certificate. However, as part of the initialization process, the initial cipher text merely includes random data in an encrypted format. Accordingly, as part of the initialization process, cryptographic block 322, following decryption of cipher text G 302, produces a random chip ID, a random private key and a random digital certificate.

[00034] Subsequently, OEM sends request 352 to key request logic 350. Representatively, key request logic 350 directs block cipher 336 to generate a key update request (R_{key}) 340. In one embodiment, key update request 340 is formed by encrypting random chip ID 240, chip secret key 250 and a pad value 332 to preserve privacy. In one embodiment a public key crypto-system is used to encrypt the information using a public key of a trusted key distribution facility, such as KDF 270 of FIG. 2.

[00035] Once the key update request is received by OEM 300, OEM 300 signs random cipher text G 302 with a private key of the OEM (K_{OEM}) to produce a digital signature (S(G)). As known to those skilled in the art, a digital signature represents a digital code that can be attached to an electronically transmitted message that uniquely identifies the sender of the message for security purposes. Once signed, OEM sends key request 322, signature S(G) and random cipher text G 302 to KDF 270, as further illustrated with reference to FIG. 5.

[00036] As illustrated with reference to FIG. 5, KDF 270 initially verifies S(G) using the OEM's public key (P_{OEM}). If the received digital signature is invalid, the request is ignored, otherwise, KDF 270, using request verification logic 272, decrypts chip secret key 250 and random chip ID within key request 322 using a private key of the key distribution facility (K_{KDF}). In one embodiment, request verification logic 272 computes chip ID 240 by decrypting chip secret key 250 using key (K_1) 234. Subsequently, KDF 270 computes manufacturing information 232 by decrypting chip ID 240 using a cryptographic block and key (k_1) 234. Representatively, manufacturer 200 also functions as KDF 270. However, a CA or other like trusted third party may perform the generation and assignment of private key 290.

[00037] Accordingly, logic 272 may verify that chip secret key 250 within key request 340 is authentic by decrypting chip secret key 250 to form chip ID 240 to derive decrypted manufacturing information and compare the manufacturing information with

the initial or original manufacturing information used to form chip ID 240. If matching information is detected, control flow is provided to key generation logic 280. Otherwise, invalid request logic 274 may invalidate trust in OEM 300 and subsequently suspend trust, pending an investigation of an attempt to obtain keys for false chips.

[00038] Assuming the OEM is trusted, key generation logic 280 computes private key (PK_c) 282. Subsequently, PK_c 282 is provided to cryptographic block 286. In one embodiment block 286 performs cipher block chaining (CBC mode) encryption using a random number or initialization vector (IV) to produce a message C. As known to those skilled in the art, cipher block chaining (CBC) is a confidential mode whose encryption features the combining (chaining) of the plain text blocks with previous cipher blocks. In one embodiment, the message C or cipher text 292 is comprised of PK_c 282, a digital key certificate and chip ID 240, which are encrypted using chip secret key 250. Once formed, cipher text 292, along with initialization vector 294, are transmitted to OEM 300.

[00039] Referring again to FIG. 2, OEM 300 stores cipher text C 292 and initialization vector 292 within off-chip persistent memory of the system. Representatively, the off-chip persistent memory is flash memory. Once the cipher text 292 and IV 294 are stored, OEM reinitializes chipset 310 by providing chipset 310 with cipher text 292 during, for example, initial system boot. Once received, key logic 320 once again decrypts cipher text C in order to form chip ID 240, a digital key certificate and the at least one private key. In one embodiment, the digital key certificate is used by chipset 310 during an authentication procedure to establish a secure authenticated channel, without disclosing the identity of chipset 310 (in those embodiments where each chip receives a unique sequence of non-unique keys, or uses an authentication protocol that does not establish identity).

[00040] As known to those skilled in the art, a digital certificate represents an attachment to an electronic message used for security purposes. Accordingly, an individual wishing to send an encrypted message applies for a digital certificate from a certificate authority (CA). As described herein, a CA is a trusted third-party organization or company that issues digital certificates used to create digital signature and public-private key pairs. Hence, attachment of a digital certificate to an encrypted message enables a recipient of the encrypted message, or cipher text, to verify that the sender of the cipher text is an authenticated, or trusted, individual. Procedural methods for implementing one or more of the above-mentioned embodiments are now described.

Operation

[00041] FIG. 6 is a flowchart illustrating a method 400 for installing a chip secret key within a manufactured chip to enable the manufactured chip to receive at least one assigned private key to enable the manufactured chip to perform an authentication procedure to establish a secure authenticated channel, in accordance with one embodiment. At process block 402, a chip secret key is programmed into a manufactured chip. At process block 422, the manufactured chip is sent to an original equipment manufacturer (OEM). Subsequently, at process block 430, at least one private key is generated for the manufactured chip according to a received key update request. In one embodiment, method 400 approximately describes private key distribution, as illustrated with reference to FIGS. 2-5.

[00042] FIG. 7 is a flowchart illustrating a method 410 for programming the chip secret key into a manufactured chip, in accordance with one embodiment. At process block 412, unique identification (ID) information is gathered for the manufactured chip. In one embodiment, the identification information includes a wafer serial number of a wafer from which the manufactured chip is formed, as well as an X,Y coordinate location of the manufactured chip within the wafer. However, those skilled in the art will recognize that identification information may be generated from a wide array of sources in order to uniquely identify the manufactured chip.

[00043] At process block 414, the identification information is encrypted using a first key to form a chip ID for the manufactured chip, for example, as illustrated with reference to FIG. 3, which depicts secret key logic 230 of manufacturer 200 of FIG. 2, in accordance with one embodiment. At process block 416, the chip ID is encrypted using a second key to form the chip secret key. Once formed, at process block 418, the chip secret key is stored within fuses of the manufactured chip. Once stored, at process block 420, selected chip fuses of the manufactured chip are blown in order to prohibit reading of the chip fuses to disclose the chip secret key.

[00044] FIG. 8 is a flowchart illustrating a method 440 for generating the at least one private key of process block 430, in accordance with one embodiment. At process block 442, a key update request is received from the system OEM. Once received, at process block 444, the key update request is authenticated. At process block 468, if the key update request is authentic, process block 470 is performed. Otherwise, the key update request is disregarded and trust of the OEM is temporarily suspended, pending an

investigation. At process block 470, cipher text including the at least one private key assigned to the manufactured chip is generated. Once generated, at process block 490, the cipher text is sent to the system OEM.

[00045] FIG. 9 is a flowchart illustrating a method 450 for authenticating the received key update request of process block 444 of FIG. 8, in accordance with one embodiment. At process block 452, a digital signature of the system OEM included within the key update request is verified. At process block 454, if the digital signature of the OEM is verified, process block 456 is performed. Otherwise, the key update request is ignored. At process block 456, the key update request is decrypted to form an alleged chip ID. At process block 458, the chip ID of the manufactured chip is compared to the alleged chip ID to verify that the chip ID matches the alleged chip ID. At process block 460, if the alleged chip ID is verified, process block 462 is performed. Otherwise, at process block 466, the received key update request is disregarded. At process block 462, the alleged chip ID is decrypted in order to form alleged chip manufacturing information (AM_c).

[00046] Referring again to FIG. 3, decryption of alleged chip ID (AID_c) using key K_1 234 should yield manufacturing information M_c 232. Accordingly, at process block 264, the alleged manufacturing information AM_c is compared to chip manufacturing information M_c. Accordingly when M_c is equal to AM_c, verification of the received key update request is complete. Once the key update request is verified, control flow returns to process block 444 of FIG. 8. Otherwise, at process block 466, the key update request is disregarded.

[00047] FIG. 10 is a flowchart illustrating a method 470 for generating the cipher text, including the at least one private key of process block 460, in accordance with one embodiment. At process block 482, a unique secret value is encrypted using the chip secret key to form a key vector. In one embodiment, the key vector includes a unique series of non-unique public/private key crypto-system keys. Hence, by using a unique series of non-unique keys, the series of keys assigned to a comprised devise can be revoked without interrupting innocent devices. For such innocent devices, such devices will continue performing authentication using the first non-revoked key in their series to continue operation. The use of non-unique keys for authentication preserves the privacy of the manufactured chip.

[00048] Accordingly, in one embodiment, the initial installation of the chip secret key enables insulation of an order of magnitude more keys that would normally be used by a conventional crypto-system using less unique bits in the chip than are required to install even one asymmetric private key pair. Referring again to FIG. 10, at process block 484, all revoked keys from the key vector are removed to form a private key vector. At process block 486, the private key vector, the chip ID and digital certificates corresponding to the vector of private keys are encrypted using the chip secret key and an initialization vector to form the cipher text. Accordingly, in some embodiments, a first non-revoked private key and its corresponding digital certificate are used to form the cipher text instead of the private key vector and corresponding digital certificates.

[00049] Referring now to FIG. 11, FIG. 11 is a flowchart illustrating a method 500 for initializing an integrated chip, including a preprogrammed chip secret key to generate a key update request in order to receive an assigned, at least one private key from a key distribution facility (KDF). At process block 502, an integrated chip within a system is initialized to generate a key update request using a preprogrammed chip secret key stored within the integrated chip. At process block 530, the key update request is transmitted to a key distribution facility.

[00050] Once transmitted, the key distribution facility will generate cipher text including at least one private key assigned to the integrated chip from the KDF. Subsequently, the integrated chip may use the private key to send a received encrypted digital content in the form of cipher text, which may be decrypted using a private key of the integrated chip once received. Accordingly, by using the assigned private key, the integrated chip is capable of forming a secure authenticated channel in order to receive protected content from content protection applications.

[00051] FIG. 12 is a flowchart illustrating a method 510 for initializing the integrated chip of process block 502 of FIG. 11, in accordance with one embodiment. At process block 512, random cipher text is provided to the integrated chip. At process block 514, the integrated chip decrypts the random cipher text using the chip secret key to form a random ID, a random key and a random digital certificate. At process block 516, the OEM requests the integrated chip to generate the key update request. In response, at process block 518, the integrated chip encrypts the random ID, the chip secret key and a pad value using a public key of the KDF to form the key update request. At process

block 520, the OEM attaches a digital certificate of the OEM and a signature of the random cipher text used in step 512 to the random cipher text.

[00052] FIG. 13 is a flowchart illustrating a method 550 for initializing an integrated chip once an assigned at least one private key is received for the integrated chip, in accordance with one embodiment. At process block 552, received cipher text is provided to the integrated chip during initial boot. Once provided to the chip, the integrated chip decrypts the received cipher text using the chip secret key to form a chip ID and the at least one private key. Subsequently, at process block 556, the integrated chip authenticates with a content protection application to receive protected content. In one embodiment, during authentication, the integrated chip also provides a received digital key certificate during the authentication protocol.

[00053] Representatively, since the digital key certificate associated with, for example, a key vector, may be shared by many platforms, the digital certificate cannot be used as a platform identity. Hence, content protection applications cannot identify the recipient of content. As such, content protection applications are able to verify that the integrated chip is an authorized recipient using the private key digital certificate. Hence, privacy is maintained by using the private key digital certificate during authentication protocols. In one embodiment, privacy is best preserved if access to received cipher text is limited to access during initial boot. Subsequently, following initial boot, access to received cipher text, including the at least one private key assigned to the chip, is disabled. However, if access to the received cipher text may not be disabled following initial boot, the integrated chip may be further requested to generate a second key update request.

[00054] Referring to FIG. 14, FIG. 14 is a flowchart illustrating a method 560 for generating a second key update request, in accordance with one embodiment. At process block 560, the received cipher text is provided to the integrated chip. Subsequently, the integrated chip is requested to generate a key update request. At process block 566, the integrated chip encrypts the chip ID, the chip secret key and a pad value using a public key of the KDF to form a second key update request. At process block 568, the second key update request is transmitted to the KDF.

[00055] As such, the KDF will generate a new private key for the integrated chip to enable integrated chip to use the private key for future authentication with content protection applications. Accordingly, the process of replacing the initially assigned at

least one private key to the integrated chip may be repeated as desired. Furthermore, this process may be repeated in order to preserve privacy of the integrated chip from applications that may be able to access the received cipher text after device initialization or initial system boot.

[00056] Accordingly, conventional systems generally install a unique asymmetric crypto-system private key within a device. Unfortunately, such private keys take more space (bits) than a symmetric secret key, which is a cost problem for integrated chips since the space required to store such asymmetric or symmetric keys is costly. Furthermore, once a device authenticates with a content protection application, user privacy is generally violated since the identity of the device is made known to the authentication application. Accordingly, by using multiple, non-unique public/private key pairs to provide privacy, implementation of such a scheme would require significantly more space to store multiple keys.

[00057] Accordingly, in one embodiment, the chip secret key enables the minimum possible number of fuse bits, such as enough to prevent a hacker from attacking the compromised device by merely guessing the information, but less information than required to store a secret key of a public/private key pair. Hence, in one embodiment, the device receives an arbitrary number of keys within a key vector. Subsequently, an identify of the device is only revealed to a trusted party that distributes keys to legitimate devices during system initialization. Hence, an identity of the device is not revealed during normal use or authentication to receive protected content.

[00058] It is to be understood that even though numerous characteristics and advantages of various embodiments of the present invention have been set forth in the foregoing description, together with details of the structure and function of various embodiments of the invention, this disclosure is illustrative only. In some cases, certain subassemblies are only described in detail with one such embodiment. Nevertheless, it is recognized and intended that such subassemblies may be used in other embodiments of the invention. Changes may be made in detail, especially matters of structure and management of parts within the principles of the embodiments of the present invention to the full extent indicated by the broad general meaning of the terms in which the appended claims are expressed.

[00059] Having disclosed exemplary embodiments and the best mode, modifications and variations may be made to the disclosed embodiments while remaining within the scope of the embodiments of the invention as defined by the following claims.